

09-29-00

A



Please type a plus sign (+) inside this box PTO/SB/05 (12/97)

Approved for use through 09/30/00. OMB 0651-0032

Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UTILITY PATENT APPLICATION TRANSMITTAL (Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No. 042390.P9021 Total Pages 27

First Named Inventor or Application Identifier Scott B. Blum

Express Mail Label No. EL627467040US



ADDRESS TO: Commissioner of Patents & Trademarks
Box Patent Application
Washington, D. C. 20231

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

1. X Fee Transmittal Form
(Submit an original, and a duplicate for fee processing)
2. X Specification (Total Pages 27)
(preferred arrangement set forth below)
 - Descriptive Title of the Invention
 - Cross References to Related Applications
 - Statement Regarding Fed sponsored R & D
 - Reference to Microfiche Appendix
 - Background of the Invention
 - Brief Summary of the Invention
 - Brief Description of the Drawings (if filed)
 - Detailed Description
 - Claims
 - Abstract of the Disclosure
3. X Drawings(s) 9 (35 USC 113) (Total Sheets 6)
4. X Oath or Declaration (Total Pages 7) signed
 - a. XX Newly Executed (Original)
 - b. Copy from a Prior Application (37 CFR 1.63(d))
(for Continuation/Divisional with Box 17 completed) (**Note Box 5 below**)
 - i. DELETIONS OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).
5. Incorporation By Reference (useable if Box 4b is checked)
The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.

6. _____ **Microfiche Computer Program (Appendix)**

7. _____ Nucleotide and/or Amino Acid Sequence Submission
(if applicable, all necessary)
a. _____ Computer Readable Copy
b. _____ Paper Copy (identical to computer copy)
c. _____ Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

8. _____ Assignment Papers (cover sheet & documents(s))
9. _____ a. 37 CFR 3.73(b) Statement (where there is an assignee)
_____ b. Power of Attorney
10. _____ English Translation Document (if applicable)
11. _____ a. Information Disclosure Statement (IDS)/PTO-1449
_____ b. Copies of IDS Citations
12. _____ Preliminary Amendment
13. XX Return Receipt Postcard (MPEP 503) (Should be specifically itemized)
14. _____ a. Small Entity Statement(s)
_____ b. Statement filed in prior application, Status still proper and desired
15. _____ Certified Copy of Priority Document(s) (if foreign priority is claimed)
16. _____ Other: _____

17. **If a CONTINUING APPLICATION**, check appropriate box and supply the requisite information:
____ Continuation ____ Divisional ____ Continuation-in-part (CIP)
of prior application No: _____

18. **Correspondence Address**

- X Customer Number or Bar Code Label 008791
(Insert Customer No. or Attach Bar Code Label here)
or
X Correspondence Address Below

NAME Edwin H. Taylor, Reg. No. 25,129

ADDRESS BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
12400 Wilshire Boulevard, Seventh Floor

CITY Los Angeles STATE California ZIP CODE 90025-1026

Country U.S.A. TELEPHONE (408) 720-8598 FAX (408) 720-9397

FEE TRANSMITTAL FOR FY 2000**TOTAL AMOUNT OF PAYMENT (\$) 690.00**

Complete if Known:

Application No. UnknownFiling Date HerewithFirst Named Inventor: Scott B. BlumGroup Art Unit UnknownExaminer Name UnknownAttorney Docket No. 042390.P9021
 Jc931 U.S. PTO
 09/671767
 09/27/00
METHOD OF PAYMENT (check one)

1. ☒ **The Commissioner is hereby authorized to charge indicated fees and credit any over payments to:**

Deposit Account Number 02-2666Deposit Account Name BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, LLP☒ **Charge Any Additional Fee Required Under 37 CFR 1.16 and 1.17**

2. ☒ **Payment Enclosed:**

☒ **Check**☐ **Money Order**☐ **Other****FEE CALCULATION****1. BASIC FILING FEE**

Large Entity		Small Entity		Fee Description	Fee Paid
Code	Fee (\$)	Code	Fee (\$)		
101	690	201	345	Utility application filing fee	<u>690.00</u>
106	310	206	155	Design application filing fee	<u> </u>
107	480	207	240	Plant filing fee	<u> </u>
108	690	208	345	Reissue filing fee	<u> </u>
114	150	214	75	Provisional application filing fee	<u> </u>
SUBTOTAL (1)					<u>\$690.00</u>

2. EXTRA CLAIM FEES

Extra Claims				Fee from below	Fee Paid
Total Claims	18	- 20** =	00	X 18.00	= 00
Independent Claims	3	- 5** =	00	X 78.00	= 00
Multiple Dependent					=

****Or number previously paid, if greater; For Reissues, see below.**

Large Entity		Small Entity		Fee Description
Code	Fee (\$)	Code	Fee (\$)	
103	18	203	9	Claims in excess of 20
102	78	202	39	Independent claims in excess of 3
104	260	204	130	Multiple dependent claim, if not paid
109	78	209	39	**Reissue independent claims over original patent
110	18	210	9	**Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) \$690.00

01/10/2000

- 1 -

PTO/SB/17 (6/99)

Patent fees are subject to annual revisions. Small Entity payments must be supported by a small entity statement, otherwise large entity fees must be paid.

See Forms PTO/SB/09-12

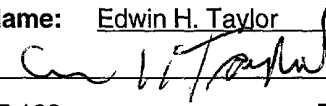
FEE CALCULATION (continued)**3. ADDITIONAL FEES**

<u>Large Entity</u>		<u>Small Entity</u>		<u>Fee Description</u>	<u>Fee Paid</u>
<u>Fee Code</u>	<u>Fee (\$)</u>	<u>Fee Code</u>	<u>Fee (\$)</u>		
105	130	205	65	Surcharge - late filing fee or oath	_____
127	50	227	25	Surcharge - late provisional filing fee or cover sheet	_____
139	130	139	130	Non-English specification	_____
147	2,520	147	2,520	For filing a request for reexamination	_____
112	920*	112	920*	Requesting publication of SIR prior to Examiner action	_____
113	1,840*	113	1,840*	Requesting publication of SIR after Examiner action	_____
115	110	215	55	Extension for response within first month	_____
116	380	216	190	Extension for response within second month	_____
117	870	217	435	Extension for response within third month	_____
118	1,360	218	680	Extension for response within fourth month	_____
128	1,850	228	925	Extension for response within fifth month	_____
119	300	219	150	Notice of Appeal	_____
120	300	220	150	Filing a brief in support of an appeal	_____
121	260	221	130	Request for oral hearing	_____
138	1,510	138	1,510	Petition to institute a public use proceeding	_____
140	110	240	55	Petition to revive unavoidably abandoned application	_____
141	1,210	241	605	Petition to revive unintentionally abandoned application	_____
142	1,210	242	605	Utility issue fee (or reissue)	_____
143	430	243	215	Design issue fee	_____
144	580	244	290	Plant issue fee	_____
122	130	122	130	Petitions to the Commissioner	_____
123	50	123	50	Petitions related to provisional applications	_____
126	240	126	240	Submission of Information Disclosure Stmt	_____
581	40	581	40	Recording each patent assignment per property (times number of properties)	_____
146	690	246	345	For filing a submission after final rejection (see 37 CFR 1.129(a))	_____
149	690	249	345	For each additional invention to be examined (see 37 CFR 1.129(a))	_____
Other fee (specify) _____					_____
Other fee (specify) _____					_____

SUBTOTAL (3) \$ 690.00

*Reduced by Basic Filing Fee Paid

SUBMITTED BY:

Typed or Printed Name: Edwin H. Taylor Customer Number: 008791
Signature  Date September 27, 2000
Reg. Number 25,129 Deposit Account User ID _____
(complete if applicable)

UNITED STATES PATENT APPLICATION

for

SUBNET INDEPENDENT TRANSPARENT BRIDGE

INVENTOR:

Scott B. Blum

File No: 42390.P9021

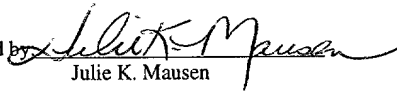
Prepared by:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN
12400 Wilshire Boulevard
Los Angeles, CA 90025-1026
(408) 720-8300

EXPRESS MAIL CERTIFICATE OF MAILING

"Express Mail" mailing label number EL627467040US Date of Deposit September 27, 2000
I hereby certify that I am causing this paper or fee to be deposited with the United States Postal Service "Express Mail Post
Office to Addressee" service on the date indicated above and that this paper or fee has been addressed to:
Box Patent Application, Assistant Commissioner for Patents, Washington, D. C. 20231.

Signed by


Julie K. Mausen

Date Signed: September 27, 2000

SUBNET INDEPENDENT TRANSPARENT BRIDGE

FIELD OF THE INVENTION

This invention relates to communication technologies generally and particularly to bridging technologies.

5

BACKGROUND OF THE INVENTION

With families bringing second and third personal computers (hereinafter PCs) into their homes, a method for the computer users in these homes to effectively utilize and share all of their computing resources is becoming increasingly desirable.

10 Home networking is one such solution, and it avails the family members with the following services, but without limitation, printer sharing, simultaneous Internet Access, file sharing, multi-player gaming, etc.

Many home networking technologies exist today, and a number of the technologies seek to take advantage of the readily available infrastructure in a home.

15 For instance, the new Home Phoneline Networking Alliance (hereinafter HomePNA) 2.0 specification, published in December of 1999, enables 10Mbps networking using a home's existing phone lines and conventional RJ-11 phone jacks. Additionally, open specifications ratified by the HomeRF™ (Radio Frequency) Working Groups utilize 2.4-GHz frequency-hopping technology to
20 enable a wireless home network.

Despite the advantages and the convenience that the aforementioned home networking technologies bring to a home, there lacks a cost-effective mechanism for these technologies to interoperate with one another. Figure 1 illustrates one prior art configuration that deploys HomePNA and HomeRF technologies. Specifically,

5 primary PC 100 accesses the Internet through its Internet access point 102. Internet access point 102 can be, but not limited to, a cable modem, a digital subscriber line (DSL) modem, a dial-up modem, etc. Then primary PC 100 communicates with second PC 112 via HomePNA Network Interface Card (hereinafter NIC) 106 and HomePNA NIC 110, respectively. Similarly, primary PC 100 communicates with
10 third PC 116 via HomeRF NIC 108 and HomeRF 114, respectively.

Assuming the communications among the NICs are based on Transmission Control Protocol/Internet Protocol (hereinafter TCP/IP), difficulties arise when the operating system on primary PC 100 does not support sharing one network prefix for HomePNA NIC 106 and HomeRF NIC 108. As a result, in order for primary PC
15 100 to communicate with second PC 112 and third PC 116 directly, HomePNA NIC 106 and HomePNA NIC 110 would be configured to share one network prefix, and HomeRF NIC 108 and HomeRF NIC 114 would be configured to share another. With this configuration, however, second PC 112 cannot communicate with third PC 116 directly. In other words, network packets originated from HomePNA NIC 110

would most likely only reach HomeRF NIC 114 through a router that shares the same network prefix as HomeRF NIC 114.

Furthermore, even though bridging technology exists today to allow communication between one type of network with another, such technology is often embodied in a standalone apparatus. Not only is modifying a hardware bridge to accommodate new technologies or specifications nontrivial, but a hardware bridge is also more costly than a software solution that utilizes available PC resources.

Therefore, a method and apparatus is needed to overcome the described operating system limitations and to enable integration between two or more different networking technologies cost effectively.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and is not limited by the figures of the accompanying drawings, in which like references indicate similar elements, and in which:

5 **Figure 1** illustrates one prior art configuration that deploys HomePNA and HomeRF technologies.

Figure 2(a) illustrates one embodiment of a subnet independent bridge that utilizes a standardized driver interface.

Figure 2(b) illustrates a functional block diagram of one embodiment of a
10 subnet independent bridge.

Figure 3 illustrates a general block diagram of a general-purpose computer system

Figure 4 illustrates an application of one embodiment of a SI bridge in a home networking configuration.

15 **Figure 5** illustrates a flow chart of one process that one embodiment of a packet delivery engine follows.

DETAILED DESCRIPTION

A method and an apparatus for intelligently bridging a first network and a second network are disclosed. In the following paragraphs, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it will be apparent to one of ordinary skill in the art that the invention may be practiced without these particular details. In other instances, well-known elements and theories, such as Transmission Control Protocol/Internet Protocol (hereinafter TCP/IP), NetBios Enhanced User Interface (hereinafter Netbeui) Protocol, Internetwork Packet Exchange (hereinafter IPX) Protocol, Institute of Electrical and Electronics Engineers (hereinafter IEEE) 802 standards, Ethernet IP addressing schemes, Peripheral Component Interconnect (hereinafter PCI), Universal Serial Bus (hereinafter USB), operating systems, kernel mode, user mode, device driver, Open Systems Interconnection (hereinafter OSI) Reference Model, Network Driver Interface Specification (hereinafter NDIS), etc. have not been discussed in special details in order to avoid obscuring the present invention.

Moreover, throughout the following discussions, a network interface card (hereinafter NIC) can be, but not limited to, a PCI or a USB expansion board, a semiconductor device that has been integrated into a computer system's motherboard and provides network connectivity functionality or a credit-card size peripheral device, etc. The term, "intermediate driver", refers to a driver that

interfaces with an upper-level driver and a lower-level driver. A protocol driver that implements layers of a network protocol stack is one example of an upper-level driver. On the other hand, a miniport driver that implements device-specific operations for a NIC is an example of a lower-level driver. Also, the terms,

- 5 “protocol stack” and “network stack”, are used interchangeably to broadly refer to a hierarchy of protocol layers or levels that provide appropriate networking functions. For example, the OSI Reference Model describes networking as a series of layers with a specific set of functions allocated to each layer. These layers are (from the lowest level layer to the highest level layer): physical, data link, network, transport,
- 10 session, presentation and application layers. Each lower-level layer offers certain services to a higher-level layer and shields the higher-level layer from the implementation details of the services. In addition, the term “network topology” broadly refers to descriptions of the accessibility of certain machines or NICs on particular networks. Finally, a machine readable medium refers to, but not limited
- 15 to, a storage device, a memory device, a carrier wave, etc.

Figure 2(a) illustrates one embodiment of present invention, a subnet independent bridge (hereinafter SI bridge). SI bridge utilizes a standardized driver interface. One example of this standardized driver interface is NDIS 208. NDIS abstracts lower-level drivers from upper-level drivers by providing the layered

20 drivers with a library of function calls and by specifying a standard interface

between the layered drivers. NDIS 208 also maintains state information and parameters for the drivers, including pointers to functions, parameter blocks for linkage, handles and other system values. The structure that utilizes the NDIS interface 208 shown in Figure 2(a) mainly operates in two levels: user level 200 and

5 kernel level 202. More particularly, application 204, such as Internet browser, operates in user level 200. SI bridge 210, NIC driver 212 and protocol driver 206, which implements the transport layer of a protocol stack, such as a TCP/IP, an IPX or a Netbeui stack, operate in kernel mode 202.

In addition to calling NDIS 208 provided functions (i.e. NdisXxx functions),

10 protocol driver 206 also provides NDIS 208 with protocol Application Programming Interface (hereinafter API) 216. Specifically, protocol driver 206 exports a set of ProtocolXxx functions that NDIS 208 can invoke either for its own purpose or on behalf of lower-level drivers, such as SI bridge 210 and NIC driver 212. In a similar fashion, NIC driver 212 communicates with NIC 214 by invoking NdisXxx

15 functions. In order to allow NDIS 208 to access NIC driver 212 either for its own purpose or on behalf of higher-level drivers, such as SI bridge 210 and protocol driver 206, NIC driver 212 also provides NDIS 208 with an API, miniport API 222 (i.e. MiniportXxx functions).

Because of its intermediate position in the hierarchy as shown in Figure 2(a),

20 SI bridge 210 provides protocol API 220 at its lower edge and miniport API 218 at

its higher edge to avoid disrupting the operations of the components in this hierarchy. More particularly, SI bridge 210's protocol API 220 permits NDIS 208 to communicate requests from NIC driver 212 to SI bridge 210. To NIC driver 212, SI bridge 210 is no different than protocol driver 206, and the inclusion of SI bridge 210 thus does not alter NIC driver 212's operations. Similarly, NDIS 208, either for its own purpose or on behalf of higher-level driver such as protocol driver 206, is able to access SI bridge 210 through miniport API 218. As a result, SI bridge 210 appears to be NIC driver 212 from the perspective of protocol driver 206, and protocol driver 206 can therefore continue to operate in a same matter. In other words, SI bridge 210 is "transparent" to both protocol driver 206 and NIC driver 212.

Moreover, Figure 2(b) illustrates a functional block diagram of one embodiment of a SI bridge. Specifically, SI bridge 210 contains API translator 224, packet analyzer 226 and packet delivery engine 228. API translator 224 is responsible for providing APIs to and interpreting requests from components that are external to SI bridge 210. Using Figure 2(a) as an illustration, API translator 224 exposes miniport API 218 and protocol API 220 to NDIS 208, protocol driver 206 and NIC driver 212. In a situation where protocol driver 206 intends to send packets or requests to NIC driver 212, NDIS 208 invokes MiniportXxx functions of miniport API 218 on behalf of protocol driver 206 to send such packets or requests to SI

bridge 210. API translator 224 interprets these function calls and invokes corresponding NDISXxx function calls to propagate these packets and requests to NIC driver 212.

- On the other hand, when NIC driver 212 intends to send up its status,
- 5 received packets or responded requests to protocol driver 206, NDIS 208 invokes ProtocolXxx functions of protocol API 220 on behalf of NIC driver 212 to send up the information. API translator 224 maps these ProtocolXxx function calls into appropriate NDISXxx function calls and invokes these mapped NDISXxx functions to relay the information to protocol driver 206.
- 10 Packet analyzer 226 and packet delivery engine 228 are responsible for monitoring and updating network topology information and intelligently delivering network packets to appropriate destinations. In one embodiment, packet analyzer 226 parses the network packets that SI bridge 210 receives, extracts appropriate information contained in the network packets and based on this extracted
- 15 information, constructs a distribution table that describes how SI bridge 210 should deliver the packets that it receives. For example, one implementation of packet analyzer 226 extracts source address X of a network packet that SI bridge 210 receives from network Z. According to the source address and network information, SI bridge 210 identifies NIC Y to be the sender of this network packet, and NIC Y is
- 20 reachable on network Z. Then packet analyzer 226 makes an entry in the

distribution table indicating that subsequent network packets that are destined for NIC Y should use network Z.

Because network topology may change as components are added, removed or powered up and down on networks, one embodiment of packet analyzer 226 also

5 records the arrival time of the received network packets in the distribution table.

Whenever the source address of the received network packet has already been noted in the distribution table, packet analyzer 226 only updates the arrival time. Thus, the time recorded in the distribution table indicates the last time that SI bridge 210

receives a network packet from a particular NIC. Then packet analyzer 226 can

10 periodically purge entries in the distribution table that have remained unchanged for

more than a predetermined amount of time. Alternatively, packet analyzer can

compile statistics of the network packets that it analyzes, such as the frequency of a

particular NIC sends a packet to SI bridge 210, and dynamically modifies its purging

mechanism based on such compiled statistics.

15 In addition to extracting source address information, one embodiment of packet analyzer 226 also parses out destination address and packet type information

for packet delivery engine 228. As has been mentioned before, packet delivery

engine 228 is responsible for forwarding network packets that SI bridge 210 has

received to suitable destinations. One embodiment of packet delivery engine 228

20 determines these suitable destinations based on the parsed information from packet

analyzer 226 and also the information contained in the aforementioned distribution table. Subsequent paragraphs will discuss this determination process in more details. Also, one embodiment of packet delivery engine 228 interfaces with API translator 224, where API translator 224 invokes the appropriate function calls, such as NDISXxx function calls, to further facilitate the delivery of network packets to the destined NIC(s).

It is important to emphasize that although the preceding paragraphs describe SI bridge 210 specifically in conjunction with a standardized driver interface, such as NDIS, it should be apparent to one with ordinary skill in the art to design SI bridge 210 to interoperate with other driver interfaces. Moreover, it should also be apparent to one ordinarily skilled in the art to combine or further divide up the described functional blocks of SI bridge 210.

Figure 3 illustrates a general block diagram of general-purpose computer system 300 that the discussed SI bridge 210 operates on. It should however be apparent to one ordinarily skilled in the art to implement SI bridge 210 on a number of hardware platforms that are different from general-purpose computer system 300 without exceeding the scope of the present invention. Some examples of the hardware platforms are, but not limited to, an Internet access point apparatus, a game console, etc.

General-purpose computer system 300 comprises microprocessor 302 and cache memory 306 coupled to each other through processor bus 304. Sample computer system 300 also includes high performance system bus 308 and standard I/O bus 328. Coupled to high performance system bus 308 are microprocessor 302 and system controller 310. Additionally, system controller 310 is coupled to memory subsystem 316 through channel 314, is coupled to I/O controller hub 326 through link 324 and is coupled to graphics controller 320 through interface 322. PCI NIC 334 and/or USB NIC 336, which provide sample computer system 300 access to a network, are also coupled to I/O controller hub 326. Coupled to graphics controller is video display 318. Coupled to standard I/O bus 328 are I/O controller hub 326, mass storage 330 and alphanumeric input device or other conventional input device 332.

These elements perform their conventional functions well known in the art. Moreover, it should have been apparent to one ordinarily skilled in the art that computer system 300 could be designed with multiple microprocessors 302 and may have more components than that which is shown. Also, mass storage 320 may be used to provide permanent storage for the executable instructions of SI bridge 210 in one embodiment. On the other hand, memory subsystem 316 may be used to temporarily store the executable instructions during execution by microprocessor 302.

Operation of one Embodiment of a SI bridge

Figure 4 illustrates an application of one embodiment of a SI bridge in a home networking configuration. Specifically, SI bridge 210 as shown in Figure 2(a) and 2(b) operates on general-purpose computer system 300 as shown in Figure 3. SI
5 bridge 210 is bound to multiple NICs, such as HomePNA NIC 406 and HomeRF NIC 410 through HomePNA NIC driver 404 and HomeRF NIC driver 408, respectively. As has been discussed in the Background section, HomePNA technology enables 10Mbps networking using convention home phone lines and RJ-
10 11 phone jacks. HomeRF technology, on the other hand, enables a wireless home network. Also, SI bridge 210 is bound to TCP/IP protocol driver 402.

Furthermore, since only HomePNA NIC 406 is bound to both SI bridge 210 and TCP/IP protocol driver 402, HomePNA NIC 406 has its own unique IP address and serves as the primary NIC for general-purpose computer system 300.
15 Consequently, the Media Access Control (hereinafter MAC) address of HomePNA NIC 406 not only becomes the source address for its network packets, but also becomes the source address for the network packets that are originated from secondary NICs, such as HomeRF NIC 410. Unlike HomePNA NIC 406, HomeRF NIC 410 is not bound to a protocol driver and does not need to have its own IP
20 address.

It is important to note that SI bridge 210 is protocol independent. In contrast to the network prefix limitation that the prior art configuration as described in the Background section and in Figure 1 faces, SI bridge 210 eliminates the limitation by not forwarding network packets based on IP addresses. As a specific example, for

5 network packets that conform to IEEE 802.3 standard, when SI bridge 210 receives such network packets, it looks at information contained in the packets, such as their source MAC address and destination MAC address information. Then, SI bridge 210 either passes the network packets up to upper layers of a protocol stack, such as TCP/IP protocol driver 402, or passes the packets laterally from one NIC, such as

10 HomePNA NIC 406, to another NIC, such as HomeRF NIC 410. Figure 4 also illustrates six paths for network traffic. They are paths 420, 422, 424, 426, 428 and 430.

Figure 5 illustrates a flow chart of one process that one embodiment of packet delivery engine 228 follows. In conjunction with Figure 4 and assuming the

15 following:

1. SI bridge 210 has already received a network packet;
2. Packet analyzer 226 has already constructed a distribution table;
3. Packet analyzer 226 has already parsed out relevant information of the received network packet; and

4. SI bridge 210 instructs NIC drivers, such as HomePNA NIC driver 404 and HomeRF NIC driver 408, to operate in a promiscuous mode. SI bridge 210 can then see network packets on every network that it is coupled to.

5 In block 500, packet delivery engine 228 verifies whether the parsed source address belongs to the designated primary NIC, or HomePNA NIC 406 in the embodiment illustrated in Figure 4. If the verification indicates that HomePNA NIC 406 has received its own network packet, this packet is dropped in block 502. Packet delivery engine 228 then waits to process the next network packet. However, if the

10 verification returns false, packet delivery engine 228 searches the aforementioned distribution table for entries that match the parsed source address in block 504. If packet delivery engine 228 could not identify any entries that match, it proceeds to add the unmatched source address information along with other associated network information to the table in block 506. "Other associated network information" here

15 includes, but not limited to, the associated NIC that generates the network packet, the network that the associated NIC attaches to, the time the network packet arrives at, etc. Either after updating the distribution table in block 506 or locating a matching entry in block 504, packet delivery engine 228 proceeds further to establish delivery paths for the network packet.

If packet delivery engine 228 determines that the network packet is a broadcast packet in block 508, where such type of packet is destined for all entities on a network, it then ensures the broadcast packet would travel on either receiving paths 420 and 424 or 430 and 426 in block 510. For example, when HomePNA NIC 412 sends out a broadcast packet, this network packet would travel via path 420 to reach the upper layers of general-purpose computer system 300's protocol stack through HomePNA NIC 406, HomePNA NIC driver 404, SI bridge 210 and TCP/IP protocol driver 402. Additionally, SI bridge 210 would also transport this broadcast packet laterally to HomeRF NIC 414 through path 424 and through HomeRF NIC driver 408 and HomeRF NIC 410. Similarly, network packets originated from HomeRF NIC 414 can reach upper layers of general-purpose computer system 300's protocol stack via path 430 and can directly reach HomePNA NIC 412 via path 426.

When the network packet is a multicast packet, instead of a broadcast packet in block 512, packet delivery engine 228 also selects receiving paths 420 and 424 or 430 and 426 for the multicast packet as has been discussed above. A multicast packet is somewhat similar to a broadcast packet, except that it is limited to certain destination addresses. If the network packet is neither a broadcast packet nor a multicast packet, packet delivery engine 228 proceeds to confirm its directed packet type status in block 514. A directed packet refers to a network packet that is destined to a specific, or directed, NIC.

For a confirmed directed packet, if packet delivery engine 228 further establishes that the directed packet is destined for a primary NIC, such as HomePNA NIC 406, in block 516, packet delivery engine 228 chooses receiving path 420 or 430 for the directed packet in block 518. For instance, when HomePNA NIC 412
5 sends this directed packet for HomePNA NIC 406, path 420 is selected to receive the packet. On the other hand, when HomeRF 414 sends out the directed packet, path 430 is chosen. If the directed packet is not destined for a primary NIC, packet delivery engine 228 instead designates either receiving path 424 or 426 in block 520.

Although specific examples have been provided to illustrate the operations of
10 a SI bridge, one with ordinary skill in the art may implement the illustrated system without all the disclosed details. For example, an ordinary artisan may implement SI bridge 210 to interoperate with more than the two NICs as shown in Figure 4 and also with other types of NICs than the ones described, such as Ethernet NICs, without exceeding the scope of the present invention. Additionally, an ordinary
15 artisan may also replace TCP/IP protocol driver 402 as shown in Figure 4 with, but not limited to, an IPX or a Netbeui protocol driver.

Thus, a method and apparatus for intelligently bridging a first network and a second network have been disclosed. Although a SI bridge has been described particularly with reference to the figures, it may appear in any number of electronic
20 apparatuses and network configurations. It is further contemplated that many

Variable	Unit	Value
Age	Years	45.0
Sex	Male/Female	50/50
Height	cm	170.0
Weight	kg	70.0
BMI	kg/m ²	24.0
Heart rate	beats/min	75.0
Blood pressure	mmHg	120/80
Cholesterol	mmol/L	5.0
Glucose	mmol/L	5.0
Hemoglobin	g/dL	15.0
Hematocrit	%	45.0
Platelets	10 ⁹ /L	150.0
White blood cells	10 ⁹ /L	7.0
Neutrophils	%	60.0
Lymphocytes	%	30.0
Monocytes	%	5.0
Eosinophils	%	2.0
Basophils	%	1.0
Prothrombin time	sec	12.0
Partial thromboplastin time	sec	30.0
Fibrinogen	g/L	3.0
D-dimer	ng/mL	0.5
C-reactive protein	mg/L	1.0
Interleukin-6	pg/mL	1.0
Tumor necrosis factor- α	pg/mL	1.0
Interleukin-10	pg/mL	1.0
Interleukin-17	pg/mL	1.0
Interleukin-21	pg/mL	1.0
Interleukin-22	pg/mL	1.0
Interleukin-23	pg/mL	1.0
Interleukin-24	pg/mL	1.0
Interleukin-25	pg/mL	1.0
Interleukin-26	pg/mL	1.0
Interleukin-27	pg/mL	1.0
Interleukin-28	pg/mL	1.0
Interleukin-29	pg/mL	1.0
Interleukin-30	pg/mL	1.0
Interleukin-31	pg/mL	1.0
Interleukin-32	pg/mL	1.0
Interleukin-33	pg/mL	1.0
Interleukin-34	pg/mL	1.0
Interleukin-35	pg/mL	1.0
Interleukin-36	pg/mL	1.0
Interleukin-37	pg/mL	1.0
Interleukin-38	pg/mL	1.0
Interleukin-39	pg/mL	1.0
Interleukin-40	pg/mL	1.0
Interleukin-41	pg/mL	1.0
Interleukin-42	pg/mL	1.0
Interleukin-43	pg/mL	1.0
Interleukin-44	pg/mL	1.0
Interleukin-45	pg/mL	1.0
Interleukin-46	pg/mL	1.0
Interleukin-47	pg/mL	1.0
Interleukin-48	pg/mL	1.0
Interleukin-49	pg/mL	1.0
Interleukin-50	pg/mL	1.0
Interleukin-51	pg/mL	1.0
Interleukin-52	pg/mL	1.0
Interleukin-53	pg/mL	1.0
Interleukin-54	pg/mL	1.0
Interleukin-55	pg/mL	1.0
Interleukin-56	pg/mL	1.0
Interleukin-57	pg/mL	1.0
Interleukin-58	pg/mL	1.0
Interleukin-59	pg/mL	1.0
Interleukin-60	pg/mL	1.0
Interleukin-61	pg/mL	1.0
Interleukin-62	pg/mL	1.0
Interleukin-63	pg/mL	1.0
Interleukin-64	pg/mL	1.0
Interleukin-65	pg/mL	1.0
Interleukin-66	pg/mL	1.0
Interleukin-67	pg/mL	1.0
Interleukin-68	pg/mL	1.0
Interleukin-69	pg/mL	1.0
Interleukin-70	pg/mL	1.0
Interleukin-71	pg/mL	1.0
Interleukin-72	pg/mL	1.0
Interleukin-73	pg/mL	1.0
Interleukin-74	pg/mL	1.0
Interleukin-75	pg/mL	1.0
Interleukin-76	pg/mL	1.0
Interleukin-77	pg/mL	1.0
Interleukin-78	pg/mL	1.0
Interleukin-79	pg/mL	1.0
Interleukin-80	pg/mL	1.0
Interleukin-81	pg/mL	1.0
Interleukin-82	pg/mL	1.0
Interleukin-83	pg/mL	1.0
Interleukin-84	pg/mL	1.0
Interleukin-85	pg/mL	1.0
Interleukin-86	pg/mL	1.0
Interleukin-87	pg/mL	1.0
Interleukin-88	pg/mL	1.0
Interleukin-89	pg/mL	1.0
Interleukin-90	pg/mL	1.0
Interleukin-91	pg/mL	1.0
Interleukin-92	pg/mL	1.0
Interleukin-93	pg/mL	1.0
Interleukin-94	pg/mL	1.0
Interleukin-95	pg/mL	1.0
Interleukin-96	pg/mL	1.0
Interleukin-97	pg/mL	1.0
Interleukin-98	pg/mL	1.0
Interleukin-99	pg/mL	1.0
Interleukin-100	pg/mL	1.0

CLAIMS

What is claimed is:

1. A method for intelligently bridging a first network and a second network, comprising:
 - a. binding a protocol independent bridge device driver to a first and a second network interface card device driver to access the first and the second network;
 - b. adaptively deriving topology of the first and the second network from network packets of the first and the second network that are received by the bridge device driver; and
 - c. delivering the received network packets based on information contained in the received network packets and the derived topology.
2. The method according to claim 1, wherein the bridge device driver is a protocol independent intermediate driver that utilizes a standardized driver interface.
3. The method according to claim 2, further comprising:
 - a. exposing a first application programming interface to the standardized driver interface and a protocol driver;

- b. exposing a second application programming interface to the standardized driver interface and the first and the second network interface card drivers; and
 - c. translating instructions from the protocol driver, the standardized driver interface and the first and the second network interface card drivers to effectuate transparency of the bridge device driver.
- 4. The method according to claim 1, further comprising:
 - generating and modifying entries of unmatched source addresses and associated network information of the received network packets in a distribution table.
- 5. The method according to claim 4, further comprising:
 - a. filtering the received network packets by comparing source addresses of the received network packets to the generated entries of unmatched source addresses; and
 - b. delivering the filtered received network packets according to their destination addresses, packet types and information in the distribution table.
- 6. The method according to claim 2, the standardized driver interface refers to Network Driver Interface Specification.

7. A machine readable medium having embodied thereon instructions of a bridge device driver, which when executed by a machine, causes the machine to intelligently bridge a first network and a second network, the instructions comprising:
 - a. accessing the first and the second network through interfacing with a first and a second network interface card driver;
 - b. adaptively deriving topology of the first and the second network from network packets that are received from the first and the second network; and
 - c. delivering the received network packets based on information contained in the received network packets and the derived topology.
8. The machine readable medium according to claim 7, wherein the bridge device driver is a protocol independent intermediate driver that utilizes a standardized driver interface.
9. The machine readable medium according to claim 8, the instructions further comprising:
 - d. exposing a first application programming interface to the standardized driver interface and a protocol driver;

- e. exposing a second application programming interface to the standardized driver interface and the first and the second network interface card drivers; and
- f. translating instructions from the protocol driver, the standardized driver interface and the first and the second network interface card drivers to effectuate transparency of the bridge device driver.

10. The machine readable medium according to claim 7, the instructions further comprising:
 - generating and modifying entries of unmatched source addresses and associated network information of the received network packets in a distribution table.
11. The machine readable medium according to claim 10, the instructions further comprising:
 - c. filtering the received network packets by comparing source addresses of the received network packets to the generated entries of unmatched source addresses; and
 - d. delivering the filtered received network packets according to their destination addresses, packet types and information in the distribution table.

12. The machine readable medium according to claim 8, the standardized driver interface refers to Network Driver Interface Specification.

13. A computer system, comprising:

- a. a first network interface card to couple to a first network and a second network interface card to couple to a second network;
- b. a system controller, coupled to a processor and coupled to an Input/Output controller hub that further couples to the first and the second network interface cards;
- c. a memory subsystem, having embodied thereon instructions for a protocol independent bridge device driver, which when executed by the processor, causes the computer system to intelligently bridge the first network and the second network by:
 - interfacing the bridge device bridge with a first and a second network interface card driver;
 - adaptively deriving topology of the first and the second network from network packets received from the first and the second network; and
 - delivering the received network packets based on information contained in the received network packets and the derived topology.

14. The computer system according to claim 13, wherein the bridge device driver is an intermediate driver that utilizes a standardized driver interface resident in the memory subsystem.
15. The computer system according to claim 14, the bridge device driver further comprises:
- an Application Programming Interface translator to
 - exposes a first application programming interface to the standardized driver interface and a protocol driver;
 - expose a second application programming interface to the standardized driver interface and the first and the second network interface card drivers;
 - and
 - translate instructions from the protocol driver, the standardized driver interface and the first and the second network interface card drivers to effectuate transparency of the bridge device driver.
16. The computer system according to claim 13, the bridge device driver further comprises:

a packet analyzer to generate and modify entries of unmatched source addresses and associated network information of the received network packets in a distribution table.

17. The computer system according to claim 16, the bridge device driver further comprises:

a packet delivery engine to

filter the received network packets by comparing source addresses of the received network packets to the generated entries of unmatched source addresses; and

deliver the filtered received network packets according to their destination addresses, packet types and information in the distribution table.

18. The computer system according to claim 14, the standardized driver interface refers to Network Driver Interface Specification.

ABSTRACT OF THE DISCLOSURE

- A method and apparatus for intelligently bridging a first network and a second network are disclosed. In one embodiment, a protocol independent bridge device driver is bound to a first and a second network interface card device driver in
- 5 order to access the first and the second network. Then the bridge device driver adaptively derives topology of the first and the second network based on network packets that it receives from the first and the second network. Lastly, the bridge device driver delivers the received network packets based on information contained in the received network packets and the derived topology.

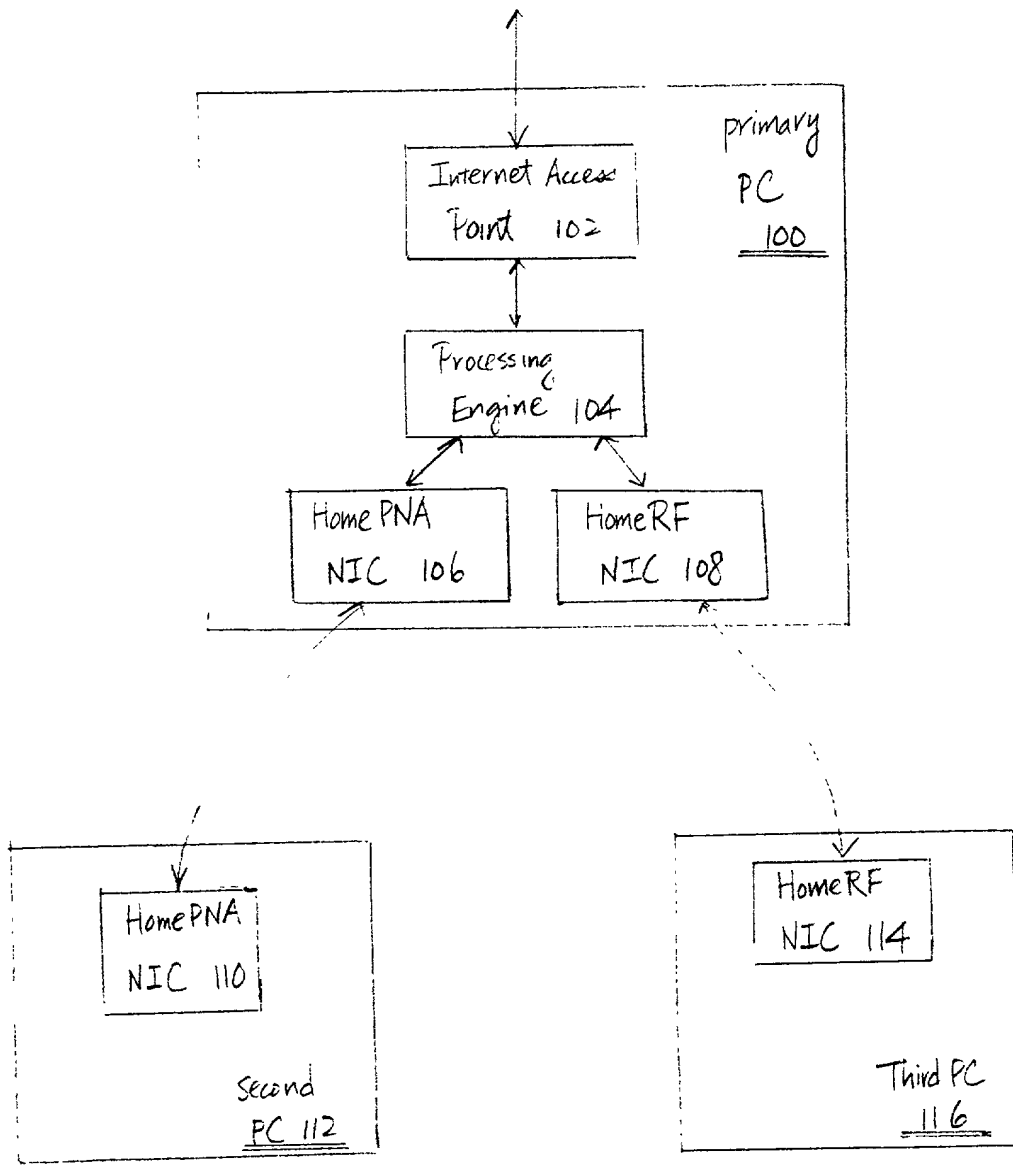


Figure 1 (prior Art)

200 202 204 206 208 210 212 214 216 218 220 222

Application
204

User
mode
200

protocol
driver 206
protocol API 216

NDIS 208

Kernel
mode
202

miniport API 218
SI bridge 210
protocol API 220

miniport API 222
NIC driver 212

NIC 214

Figure 2(a)

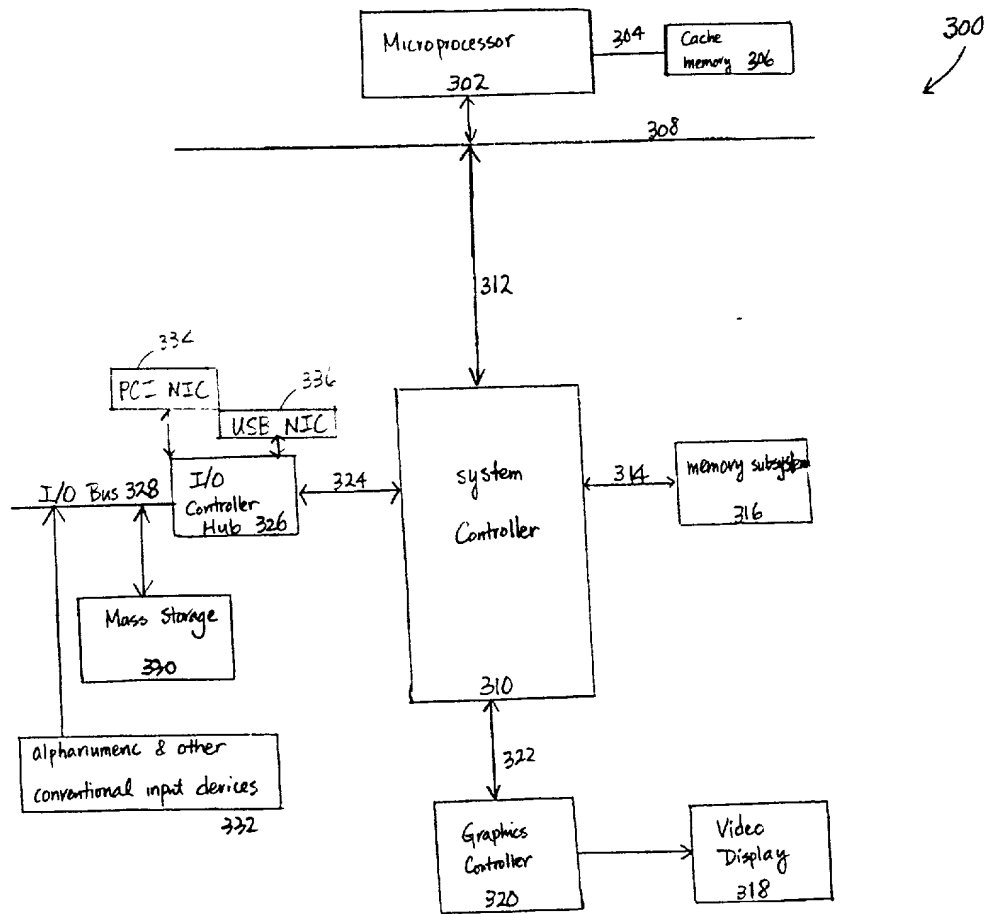


Figure 3

420
↑



Figure 4

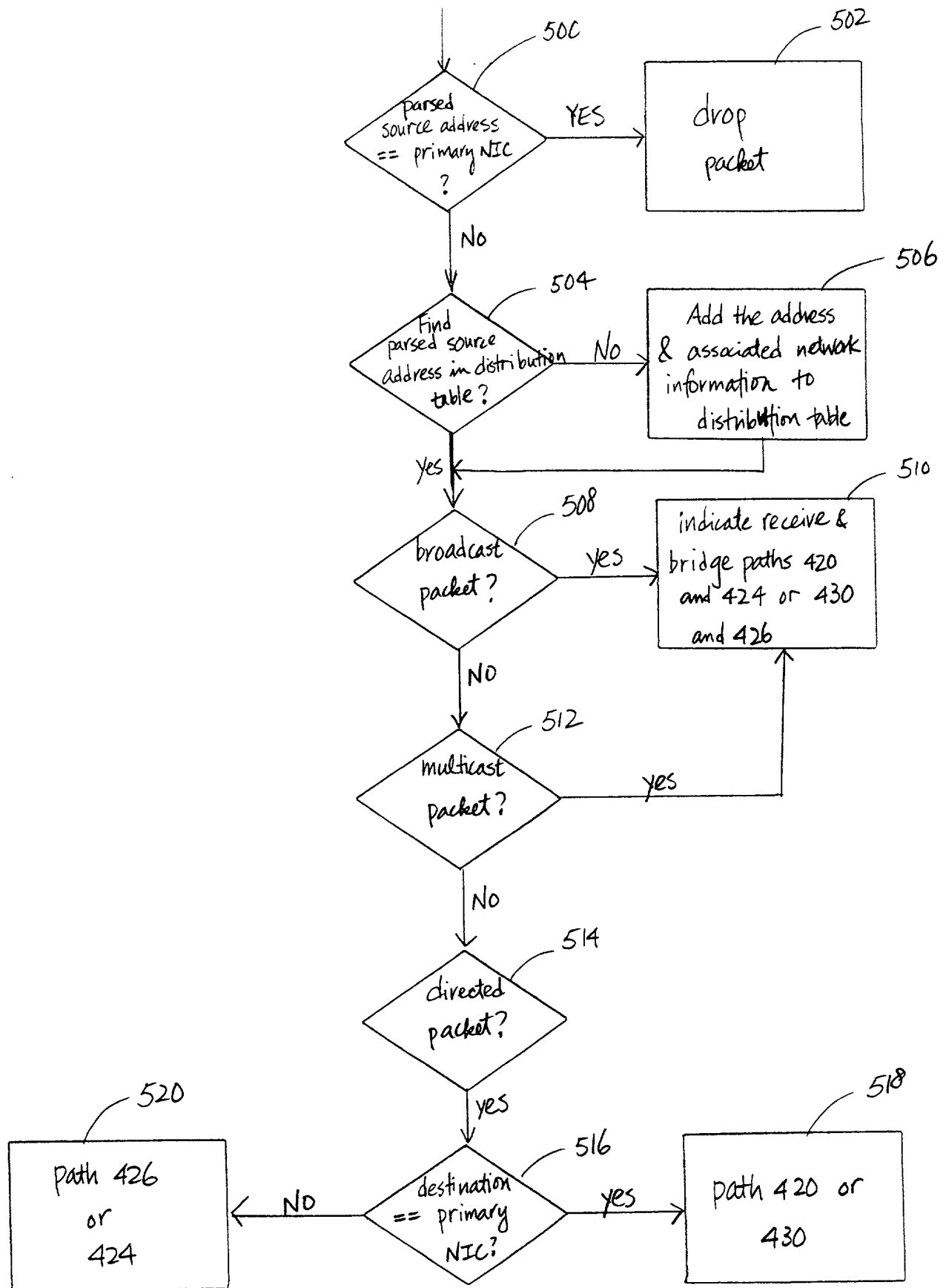
[illegible]

Figure 5

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION
(FOR INTEL CORPORATION PATENT APPLICATIONS)

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below, next to my name.

I believe I am the original, first, and sole inventor (if only one name is listed below) or an original, first, and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

SUBNET INDEPENDENT TRANSPARENT BRIDGE

the specification of which

xx is attached hereto.
 was filed on _____ as
United States Application Number _____
or PCT International Application Number _____
and was amended on _____
(if applicable)

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claim(s), as amended by any amendment referred to above. I do not know and do not believe that the claimed invention was ever known or used in the United States of America before my invention thereof, or patented or described in any printed publication in any country before my invention thereof or more than one year prior to this application, that the same was not in public use or on sale in the United States of America more than one year prior to this application, and that the invention has not been patented or made the subject of an inventor's certificate issued before the date of this application in any country foreign to the United States of America on an application filed by me or my legal representatives or assigns more than twelve months (for a utility patent application) or six months (for a design patent application) prior to this application.

I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, Section 119(a)-(d), of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

I hereby appoint the persons listed on Appendix A hereto (which is incorporated by reference and a part of this document) as my respective patent attorneys and patent agents, with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.

Send correspondence to Julie Mause, BLAKELY, SOKOLOFF, TAYLOR &
(Name of Attorney or Agent)
ZAFMAN LLP, 12400 Wilshire Boulevard 7th Floor, Los Angeles, California 90025 and direct telephone calls to ____
, (408) 720-8300.
(Name of Attorney or Agent)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Sole/First Inventor Scott B. Blum

Inventor's Signature Scott B. Blum Date 9/15/00

Residence Beaverton, Oregon Citizenship U.S.A.
(City, State) (Country)

Post Office Address 14669 NW Forest Loop
Beaverton, OR 97006

Full Name of Second/Joint Inventor _____

Inventor's Signature _____ Date _____

Residence _____ Citizenship _____
(City, State) (Country)

Post Office Address _____

Full Name of Third/Joint Inventor _____

Inventor's Signature _____ Date _____

Residence _____ Citizenship _____
(City, State) (Country)

Post Office Address _____

APPENDIX A

William E. Alford, Reg. No. 37,764; Farzad E. Amini, Reg. No. P42,261; Aloysius T. C. AuYeung, Reg. No. 35,432; William Thomas Babbitt, Reg. No. 39,591; Carol F. Barry, Reg. No. 41,600; Jordan Michael Becker, Reg. No. 39,602; Bradley J. Bereznak, Reg. No. 33,474; Michael A. Bernadicou, Reg. No. 35,934; Roger W. Blakely, Jr., Reg. No. 25,831; Gregory D. Caldwell, Reg. No. 39,926; Ronald C. Card, Reg. No. 44,587; Andrew C. Chen, Reg. No. 43,544; Thomas M. Coester, Reg. No. 39,637; Alin Corie, Reg. No. P46,244; Dennis M. deGuzman, Reg. No. 41,702; Stephen M. De Klerk, under 37 C.F.R. § 10.9(b); Michael Anthony DeSanctis, Reg. No. 39,957; Daniel M. De Vos, Reg. No. 37,813; Robert Andrew Diehl, Reg. No. 40,992; Sanjeet Dutta, Reg. No. P46,145; Matthew C. Fagan, Reg. No. 37,542; Tarek N. Fahmi, Reg. No. 41,402; Paramita Ghosh, Reg. No. 42,806; James Y. Go, Reg. No. 40,621; James A. Henry, Reg. No. 41,064; Willmore F. Holbrow III, Reg. No. P41,845; Sheryl Sue Holloway, Reg. No. 37,850; George W. Hoover II, Reg. No. 32,992; Eric S. Hyman, Reg. No. 30,139; William W. Kidd, Reg. No. 31,772; Sang Hui Kim, Reg. No. 40,450; Eric T. King, Reg. No. 44,188; Erica W. Kuo, Reg. No. 42,775; Kurt P. Leyendecker, Reg. No. 42,799; Michael J. Mallie, Reg. No. 36,591; Andre L. Marais, under 37 C.F.R. § 10.9(b); Paul A. Mendonsa, Reg. No. 42,879; Darren J. Milliken, Reg. No. 42,004; Lisa A. Norris, Reg. No. 44,976; Chun M. Ng, Reg. No. 36,878; Thien T. Nguyen, Reg. No. 43,835; Thinh V. Nguyen, Reg. No. 42,034; Dennis A. Nicholls, Reg. No. 42,036; Daniel E. Ovanezian, Reg. No. 41,236; Marina Portnova, Reg. No. P45,750; Babak Redjaian, Reg. No. 42,096; William F. Ryann, Reg. No. 44,313; James H. Salter, Reg. No. 35,668; William W. Schaal, Reg. No. 39,018; James C. Scheller, Reg. No. 31,195; Jeffrey Sam Smith, Reg. No. 39,377; Maria McCormack Sobrino, Reg. No. 31,639; Stanley W. Sokoloff, Reg. No. 25,128; Judith A. Szepesi, Reg. No. 39,393; Vincent P. Tassinari, Reg. No. 42,179; Edwin H. Taylor, Reg. No. 25,129; John F. Travis, Reg. No. 43,203; George G. C. Tseng, Reg. No. 41,355; Joseph A. Twarowski, Reg. No. 42,191; Lester J. Vincent, Reg. No. 31,460; Glenn E. Von Tersch, Reg. No. 41,364; John Patrick Ward, Reg. No. 40,216; Mark L. Watson, Reg. No. P46,322; Thomas C. Webster, Reg. No. P46,154; Charles T. J. Weigell, Reg. No. 43,398; Kirk D. Williams, Reg. No. 42,229; James M. Wu, Reg. No. 45,241; Steven D. Yates, Reg. No. 42,242; and Norman Zafman, Reg. No. 26,250; my patent attorneys, and Justin M. Dillon, Reg. No. 42,486; my patent agent, of BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP, with offices located at 12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025, telephone (310) 207-3800, and Alan K. Aldous, Reg. No. 31,905; Robert D. Anderson, Reg. No. 33,826; Joseph R. Bond, Reg. No. 36,458; Richard C. Calderwood, Reg. No. 35,468; Jeffrey S. Draeger, Reg. No. 41,000; Cynthia Thomas Faatz, Reg. No. 39,973; Sean Fitzgerald, Reg. No. 32,027; John N. Greaves, Reg. No. 40,362; Seth Z. Kalson, Reg. No. 40,670; David J. Kaplan, Reg. No. 41,105; Charles A. Mirho, Reg. No. 41,199; Leo V. Novakoski, Reg. No. 37,198; Naomi Obinata, Reg. No. 39,320; Thomas C. Reynolds, Reg. No. 32,488; Kenneth M. Seddon, Reg. No. 43,105; Mark Seeley, Reg. No. 32,299; Steven P. Skabrat, Reg. No. 36,279; Howard A. Skaist, Reg. No. 36,008; Steven C. Stewart, Reg. No. 33,555; Raymond J. Werner, Reg. No. 34,752; Robert G. Winkle, Reg. No. 37,474; and Charles K. Young, Reg. No. 39,435; my patent attorneys, and Thomas Raleigh Lane, Reg. No. 42,781; Calvin E. Wells, Reg. No. P43,256; Peter Lam, Reg. No. 44,855; and Gene I. Su, Reg. No. 45,140; my patent agents, of INTEL CORPORATION; and James R. Thein, Reg. No. 31,710, my patent attorney; with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.

[illegible]